

粒子ベースボリュームレンダリングを用いた大規模 遠隔可視化の評価(3.2 第6回情報シナジー研究会, 3. 研究活動報告)

著者	櫻井 健一, 坂本 尚久, 江原 康生, 曽根 秀昭 , 小山田 耕二
雑誌名	年報
巻	7
ページ	94-99
発行年	2008-07
URL	http://hdl.handle.net/10097/48537

粒子ベースボリュームレンダリングを用いた 大規模遠隔可視化の評価

櫻井 健一¹, 坂本 尚久^{2,3}, 江原 康生⁴, 曾根 秀昭⁵, 小山田 耕二³

¹ 東北大学大学院 情報科学研究科

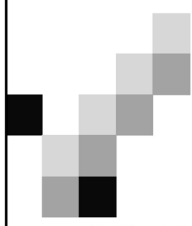
² 株式会社ケー・ジー・ティー

³ 京都大学 高等教育研究開発推進センター

⁴ 京都大学 学術情報メディアセンター

⁵ 東北大学 情報シナジーセンター ネットワーク研究部

あらまし 近年, 遠隔地に点在するスーパーコンピュータやPC クラスタなどの高い処理能力を持つ計算機をネットワークを介して有効活用し, 可視化を行う遠隔可視化への要求が高まっているが, 大規模ボリュームデータを対象とした十分な遠隔可視化手法は確立されていない. 本論文では, 効率的な大規模ボリュームデータの遠隔可視化を実現するために, 粒子ベースボリュームレンダリングを用いた遠隔可視化手法を提案し, 実際の遠隔地間ネットワーク上で提案手法の有効性を実験的に評価する.



粒子ベースボリュームレンダリングを用いた大規模遠隔可視化の評価

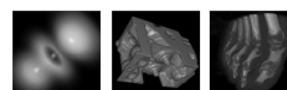
櫻井健一 *, 坂本尚久 **, 江原康生 ††, 曾根秀昭 †, 小山田耕二 †

* 東北大学大学院情報科学研究科
 ** 株式会社ケー・シー・ディー
 † 京都大学高等教育研究開発推進センター
 †† 京都大学学術情報メディアセンター
 ‡ 東北大学情報シナジーセンター

1

遠隔可視化とは(1)

- 可視化 (ボリューム可視化) とは
 - ボリュームデータの可視化
 - シミュレーションや計測の結果から生成される数値データ
 - 理解や解析を支援する有効な手段



内部構造を考慮した3次元画像

➡ ボリュームデータの大規模化

2

遠隔可視化とは(2)

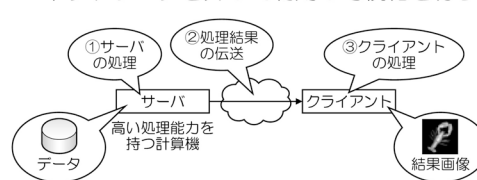
- 大規模ボリュームデータの可視化
 - 膨大な計算量とメモリが必要
 - 高い処理能力を持つ計算機を利用
 - スーパーコンピュータ (並列計算機)
 - PCクラス (複数の計算機)

➡ 高い処理能力を持つ計算機はどこにでもある訳ではない

3

遠隔可視化とは(3)

- (大規模) 遠隔可視化
 - 遠隔地に点在する計算機をネットワークを介して利用し可視化を行う



4

既存のシステムの問題点(1)

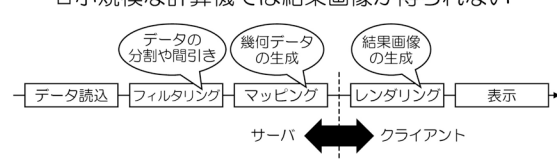
- 大規模遠隔可視化への要求
 - 遠隔地間ネットワーク上での大規模ボリュームデータの遠隔可視化
 - ネットワーク的に離れた地点間
 - サーバが計算量とメモリを要する処理を行う
 - クライアントは小規模な計算機で十分とする

➡ 既存の可視化手法を用いた遠隔可視化システムは不十分

5

既存のシステムの問題点(2)

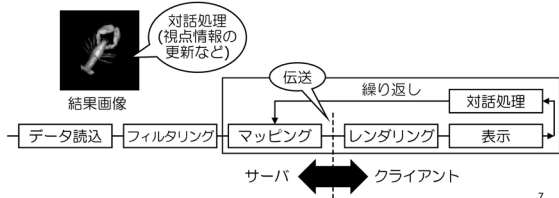
- クライアントの処理の負荷
 - 順序付けがボトルネック
 - 計算量とメモリ, グラフィックス性能を要求
 - 小規模な計算機では結果画像が得られない



6

既存のシステムの問題点(3)

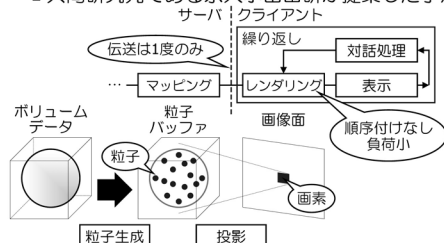
- 継続的なバーストトラフィックの発生
 - 対話処理に応じてサーバクライアント間の伝送が必要



7

提案する遠隔可視化システム(1)

- 粒子ベースボリュームレンダリング
 - 共同研究先である京大小山田研が提案した手法



8

提案する遠隔可視化システム(2)

- 粒子ベースボリュームレンダリング
 - 既存のシステムの問題点を解決
 - 大規模遠隔可視化に適した可視化手法
- 粒子ベースボリュームレンダリングを用いた遠隔可視化システムを構築し、評価
 - 実際の遠隔地間ネットワーク上で評価実験を行った

9

実験環境(1)

- ネットワークと計算機
 - 仙台京都間 / SINET3のVPN
 - クライアント@仙台
 - P4 2.8GHz, メモリ3GB
 - サーバ@京都
 - Xeon 1.6GHz (2コア) * 2 (4CPUs), メモリ8GB

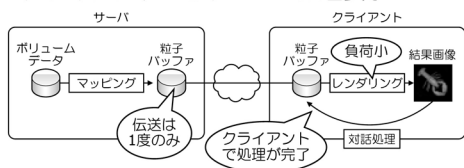
	RTT [ms]	スループット [Mbps]
参考値		
東北大内LAN	0.23	617
仙台京都間	28.86	548

RTT及びパケットロス率はping, スループットはnetperfによる測定値

10

実験環境(2)

- 実験システム
 - 可視化ライブラリKVSを利用して構築
 - Linux上で動作するPythonのプログラム
 - クライアントからサーバへ処理要求



11

実験のパラメータ(1)





- 対象ボリュームデータ
 - ボリュームデータの規模=格子数に比例
 - Hydrogen (64³), 比較的小規模なデータ
 - Bucky (512³), 大規模データ

ボリュームデータ	格子数	サイズ [byte]	結果画像
Hydrogen	64 ³	490K	
Bucky	512 ³	134M	

12

実験のパラメータ(2)

- 発生させる粒子数
 - $10^4 \sim 10^7$ まで変化させ実験
 - サーバとクライアントの処理時間, データ伝送時間を測定

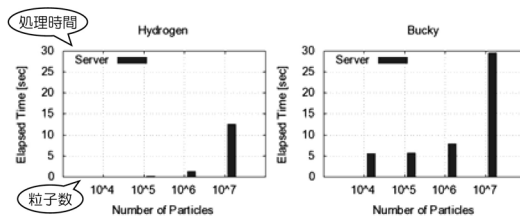
粒子数	10^4	10^5	10^6	10^7
可視化結果				

粒子数と精細さはトレードオフ

13

実験結果(1)

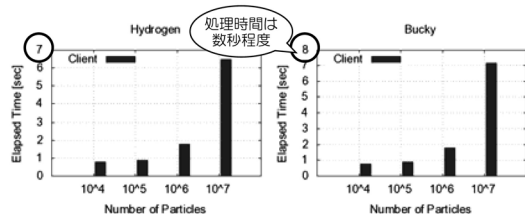
- サーバの処理結果
 - 処理時間は格子数と粒子数に比例



14

実験結果(2)

- クライアントの処理結果
 - 許容できる時間内に結果画像が得られた

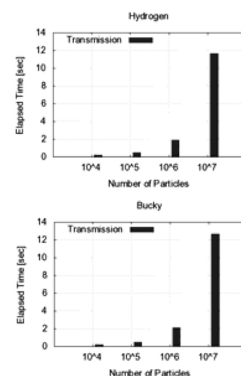


15

実験結果(3)

- データ伝送時間
 - TCPを用いて伝送
 - 外れ値を除去
 - 共用のため揺らぎ大
 - 伝送されるデータ量

粒子数	データ量 [byte]
10^4	270K
10^5	2.7M
10^6	27M
10^7	270M



16

実験結果のまとめ

- サーバの処理時間
 - 処理時間は格子数と粒子数に比例
- クライアントの処理時間
 - 許容できる時間内に結果画像が得られた
 - 処理時間は粒子数に比例
- データ伝送時間
 - 伝送ははじめの1度のみ発生
 - 伝送されるデータ量は粒子数に比例

17

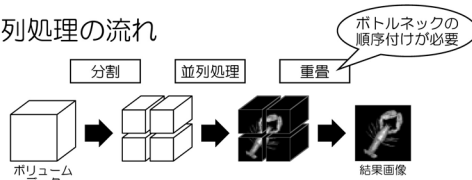
実験結果の検討

- 既存のシステムの大規模遠隔可視化における問題点を解決
 - 実験では結果画像が得られるまで数秒～数十秒を要した
 - サーバの処理時間とデータ伝送時間が大きい
 - より大規模なデータの処理に向けそれぞれの処理の効率化を図る
 - サーバの処理の並列化
 - データの並列伝送

18

サーバの処理の並列化(1)

■ 並列処理の流れ



□既存の手法, システムでは重畳の際にもボトルネックである順序付けが必要

- 大きいデータを分けて処理しても処理結果を合わせる部分が問題

19

サーバの処理の並列化(2)

■ 粒子ベースボリュームレンダリングを用いた遠隔可視化システムのサーバの処理の並列化

- 順序付けが必要ないため重畳処理がボトルネックになることはない
 - 単純に処理結果を加算すればよい
- 並列処理を行う各スレーブはそれぞれ独立に処理を行うことが可能
 - スレーブ数に比例した並列効果を期待できる

20

データの並列伝送

■ データ伝送の効率化

- 遠隔可視化のデータ伝送は損失を許容しない
 - 遠隔地間ネットワーク上では損失や輻輳の影響が大きくなる
 - UDPベースの伝送プロトコルは望ましくない
- TCPを用いた伝送の効率化
 - 輻輳制御とウィンドウサイズの変更は他のトラフィック, アプリケーションに影響
- TCPを用いた並列伝送

21

実験のパラメータ

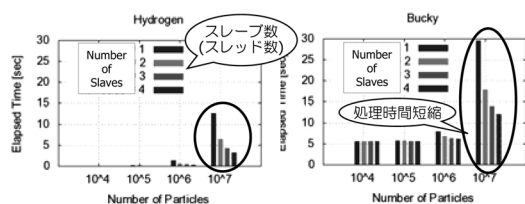
- 粒子数を $10^4 \sim 10^7$ まで変化させ実験
- サーバの処理の並列化
 - 並列計算機上でスレーブ (スレッド) 数を1~4まで変化させ処理時間を測定
- データの並列伝送
 - ストリーム数を1~50まで変化させデータ伝送時間を測定

22

サーバの処理の並列化の実験結果

■ サーバの処理時間

□スレーブ数を大きくするにつれ処理時間短縮

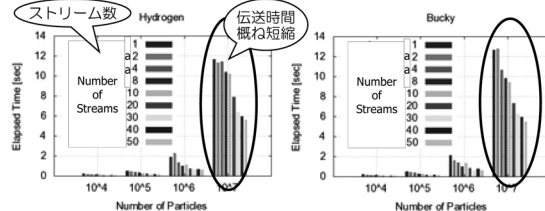


23

データの並列伝送の実験結果

■ データ伝送時間

□ストリーム数を大きくするにつれ伝送時間短縮



24

実験のまとめ

- サーバの処理の並列化
 - スレーブ数を1~4と大きくするにつれ処理時間が短縮
 - 処理時間は並列処理をしない場合の約半分に
 - どこまでスレーブ数を大きくすることで並列効果が得られるかの調査が求められる
- データの並列伝送
 - ストリーム数を1~50と大きくするにつれデータ伝送時間が短縮

25

まとめ(1)

- 粒子ベースボリュームレンダリングを用いた遠隔可視化システム
 - 既存のシステムの問題点を解決
 - クライアントの処理の負荷が大きい
 - 継続的なバーストラフィックの発生
- 実際の遠隔地間ネットワーク上でシステムを構築し, 評価実験を行った

26

まとめ(2)

- 評価実験の結果
 - クライアントは小規模な計算機でも十分
 - トラフィックの発生を最低限に抑えた
- 提案システムの有効性を確認できた
- さらに処理の効率化を試みた
 - サーバの処理の並列化
 - データの並列伝送
 - 実用的な時間 (数秒~十数秒程度) で大規模遠隔可視化の結果が得られた

27